Proceedings of Science and Mathematics
Faculty of Science, Universiti Teknologi Malaysia

Vol. 7, 2022, Page 5 - 9

Static Watson-Crick Grammars and Their Computational Properties

Aqilahfarhana Abdul Rahman[a]*, Wan Heng Fong[a], Nor Haniza Sarmin[a], Sherzod Turaev[b]

[a]Department of Mathematical Sciences, Faculty of Science,
Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia

[b]Department of Computer Science & Engineering, College of
Information Technology,United Arab Emirates University,
Al Ain 15551, United Arab Emirates.

*Corresponding author: aqilahfarhana2@graduate.utm.my

Abstract

In DNA computing, there are different computational models which are based on operations of DNA molecules and the concept of formal language theory. Watson-Crick automaton is a mathematical model that represents the biological properties of DNA; whereas a sticker system is another DNA computing model which uses the sticker operation to form complete double-stranded sequences. These two modelings of DNA molecules in DNA computing use the feature of Watson-Crick complementarity. On the other hand, Watson-Crick grammars, which are the grammar counterparts of Watson-Crick automata, use the Watson-Crick complementarity. Hence, Watson-Crick grammars and sticker systems are used in this research to propose a novel theoretical model known as static Watson-Crick grammars. A static Watson-Crick grammar generates both stranded strings dependently by checking for the Watson-Crick complementarity for each complete substring. In this research, the computational properties of static Watson-Crick grammars are investigated to correlate with the hierarchy between the families of Chomsky languages and Watson-Crick languages.

Keywords: DNA computing; Watson-Crick grammars; Sticker system; Computational properties

Introduction

DNA computing is a new discipline of research that incorporates biological molecules as the fundamental components of computing devices. In DNA computing, Watson-Crick complementarity and massive parallelism are two basic aspects required in executing DNA computations. Sticker systems and Watson-Crick automata are computational models based on different operations of DNA molecules [1]: sticker system is a language generating device that is based on the sticking operation [2], and Watson-Crick automata is an extension of finite automata with the addition of two reading heads on double sequences [3].

It has been shown in past research that formal language may be utilized to examine molecular processes from the computational perspective [4, 5]. The grammar models, however, did not use the fundamental feature of Watson-Crick complementarity of DNA molecules. The development of the Watson-Crick automata in the grammatical area started in 2012 when Watson-Crick complementarity of DNA molecules is implemented in Watson-Crick regular grammars [6]. Following that, the research has been extensively studied with Watson-Crick grammars [7] which generate double-stranded strings dynamically: the Watson-Crick complementarity can only be checked upon generating both strands of a complete double-stranded string. Hence, this research aims to introduce a new variant of Watson-Crick grammars known as static Watson-Crick grammars which use the concept of Watson-Crick grammars and sticker system; as well as to determine the computational properties in term of the Chomsky hierarchy.

Preliminaries

The reader may refer to [8, 9] for detailed information regarding the basic concepts of formal language theory, and sticker systems. Here, some notations and definitions used in this research are recalled.

Firstly, the symbol $\subseteq$ denotes the inclusion while $\subset$ denotes the strict (proper) inclusion. The membership of an element to a set is denoted by $\in$. An alphabet is a finite nonempty set of symbols. Let T be a finite alphabet. Then, T^* is the set of all finite strings (or words) over T. A string with no symbols, known as empty string, is denoted by $\boldsymbol{\lambda}$. The symbol T^+ is defined as the set of all nonempty finite strings over T where $T^+ = T^* - \{\lambda\}$. In sticker system, the set of incomplete molecules is denoted as $W_\rho(V) = L_\rho(V) \cup R(V) \cup LR_\rho(V)$, where:

$$L_\rho(V) = \left(\binom{\lambda}{V^*} \cup \binom{V^*}{\lambda}\right)\left[\begin{matrix}V\\V\end{matrix}\right]_\rho^*, R_\rho(V) = \left[\begin{matrix}V\\V\end{matrix}\right]_\rho^*\left(\binom{\lambda}{V^*} \cup \binom{V^*}{\lambda}\right) \text{ and} \tag{1}$$

$$LR_\rho(V) = \left(\binom{\lambda}{V^*} \cup \binom{V^*}{\lambda}\right)\left[\begin{matrix}V\\V\end{matrix}\right]_\rho^+\left(\binom{\lambda}{V^*} \cup \binom{V^*}{\lambda}\right). \tag{2}$$

In order to introduce the static Watson-Crick grammars, another notion of $LR_\rho(V)$ is also introduced in this research, where

$$LR_\rho^*(T) = \left(\binom{\lambda}{T^*} \cup \binom{T^*}{\lambda}\right)\left[\begin{matrix}T\\T\end{matrix}\right]_\rho^*\left(\binom{\lambda}{T^*} \cup \binom{T^*}{\lambda}\right), \tag{3}$$

$$LR_\rho^+(T) = \left(\binom{\lambda}{T^*} \cup \binom{T^*}{\lambda}\right)\left[\begin{matrix}T\\T\end{matrix}\right]_\rho^+\left(\binom{\lambda}{T^*} \cup \binom{T^*}{\lambda}\right), \tag{4}$$

and the alphabet V which is defined in $W_\rho(V)$ is used instead of the alphabet T according to the concept of grammar. Next, the definitions of Watson-Crick grammars are presented as follows.

Definition 1 [7]

A Watson-Crick (WK) grammar $G = (N, T, \rho, S, P)$ is called

- *regular* if each production has the form $A \to \langle u/v\rangle$ where $A, B \in N$ and $\langle u/v\rangle \in \langle T^*/T^*\rangle$.
- *linear* if each production has the form $A \to \langle u_1/v_1\rangle B\langle u_2/v_2\rangle$ or $A \to \langle u/v\rangle$ where $A, B \in N$ and $\langle u/v\rangle, \langle u_1/v_1\rangle, \langle u_2/v_2\rangle \in \langle T^*/T^*\rangle$.
- *context-free* if each production has the form $A \to \alpha$ where $A \in N$ and $\alpha \in (N \cup \langle T^*/T^*\rangle)^*$.

The notation $\langle u/v\rangle$ represents the element $(u, v) \subseteq V \times V$ in the set of pairs of strings and $\langle T^*/T^*\rangle$ is written instead of $V^* \times V^*$.

Results and Discussion

In this section, the definitions of static Watson-Crick regular grammars (right-linear or left-linear), static Watson-Crick linear grammars and static Watson-Crick context-free grammars are introduced.

Definition 2

A static Watson-Crick *right-linear* grammar is a 5-tuple $G = (N, T, \rho, S, P)$ where N, T are disjoint nonterminal and terminal alphabets, respectively, $\rho \in T \times T$ is a symmetric relation, $S \in N$ is a start symbol (axiom) and P is a finite set of production rules in the form of

(i) $S \to \left[\begin{matrix}\lambda\\\lambda\end{matrix}\right]$ where $S \in N$; or (5)

(ii) $S \to A\binom{x}{y}\left[\begin{matrix}u\\v\end{matrix}\right]$ where $A \in N - \{S\}$, $\binom{x}{y}\left[\begin{matrix}u\\v\end{matrix}\right] \in L_\rho(T)$; (6)

(iii) $A \to B\binom{x}{y}$ where $A, B \in N - \{S\}$, $\binom{x}{y} \in LR_\rho{}^*(T)$; or (7)

(iv) $A \to \left[\begin{matrix}u\\v\end{matrix}\right]\binom{x}{y}$ where $A \in N - \{S\}$, $\left[\begin{matrix}u\\v\end{matrix}\right]\binom{x}{y} \in R_\rho(T)$. (8)

Definition 3

A static Watson-Crick *left-linear* grammar is a 5-tuple $G = (N, T, \rho, S, P)$ where N, T are disjoint nonterminal and terminal alphabets, respectively, $\rho \in T \times T$ is a symmetric relation, $S \in N$ is a start symbol (axiom) and P is a finite set of production rules in the form of:

(i) $S \to \begin{bmatrix} \lambda \\ \lambda \end{bmatrix}$ where $S \in N$; or (9)

(ii) $S \to \begin{bmatrix} u \\ v \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} A$ where $A \in N - \{S\}$, $\begin{bmatrix} u \\ v \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \in R_\rho(T)$; (10)

(iii) $A \to \begin{pmatrix} x \\ y \end{pmatrix} B$ where $A, B \in N - \{S\}$, $\begin{pmatrix} x \\ y \end{pmatrix} \in LR_\rho{}^*(T)$; or (11)

(iv) $A \to \begin{pmatrix} x \\ y \end{pmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$ where $A \in N - \{S\}$, $\begin{pmatrix} x \\ y \end{pmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \in L_\rho(T)$. (12)

Definition 4

A static Watson-Crick *linear* grammar is a 5-tuple $G = (N, T, \rho, S, P)$ where N, T are disjoint nonterminal and terminal alphabets, respectively, $\rho \in T \times T$ is a symmetric relation, $S \in N$ is a start symbol (axiom) and P is a finite set of production rules in the form of

(i) $S \to \begin{bmatrix} \lambda \\ \lambda \end{bmatrix}$ where $S \in N$; or (13)

(ii) $S \to \begin{bmatrix} u_1 \\ v_1 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} A \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix}$ where $A \in N - \{S\}$ and $\begin{bmatrix} u_1 \\ v_1 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \in R_\rho(T), \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \in L_\rho(T)$; (14)

(iii) $A \to \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} B \begin{pmatrix} x_2 \\ y_2 \end{pmatrix}$ where $A, B \in N - \{S\}$ and $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \in LR_\rho{}^*(T)$; or (15)

(iv) $A \to \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$ where $A \in N - \{S\}$ and $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \in LR_\rho{}^*(T)$. (16)

Definition 5

A static Watson-Crick *context-free* grammar is a 5-tuple $G = (N, T, \rho, S, P)$ where N, T are disjoint nonterminal and terminal alphabets, respectively, $\rho \in T \times T$ is a symmetric relation, $S \in N$ is a start symbol (axiom) and P is a finite set of production rules in the form of

(i) $S \to \begin{bmatrix} \lambda \\ \lambda \end{bmatrix}$ where $S \in N$; or (17)

(ii) $S \to x_1 A_1 x_2 A_2 \cdots x_k A_k x_{k+1}$ where $A_i \in N - \{S\}$ for $1 \le i \le k$, $x_1 \in R_\rho(T)$, $x_i \in LR_\rho{}^+(T)$ for $2 \le i \le k$ and $x_{k+1} \in L_\rho(T)$; (18)

(iii) $A \to y_1 B_1 y_2 B_2 \cdots y_t B_t y_{t+1}$ where $A, B_i \in N - \{S\}$ for $1 \le i \le t$, $y_i \in LR_\rho{}^+(T)$ for $2 \le i \le t$, $y_1, y_{t+1} \in LR_\rho{}^*(T)$; or (19)

(iv) $A \to x$ where $A \in N - \{S\}$ and $x \in LR_\rho{}^*(T)$. (20)

The reflexive and transitive closure of $\underset{G}{\Rightarrow}$ or $(\Rightarrow)$ is denoted by $\underset{G}{\Rightarrow}^*$ or $(\Rightarrow^*)$. The language generated by static Watson-Crick grammar G, denoted by $L(G)$, is defined as

$$L(G) = \left\{u : \begin{bmatrix} u \\ v \end{bmatrix} \in WK_\rho(T) \text{ and } S \underset{G}{\Rightarrow}^* \begin{bmatrix} u \\ v \end{bmatrix}\right\}. \quad (21)$$

The family of languages generated by static Watson-Crick regular, linear, and context-free grammar is denoted by **SREG, SLIN** and **SCF**, respectively.

Next, the relationships between the families of static Watson-Crick grammars and the families in the Chomsky hierarchy and Watson-Crick grammars are determined.

Theorem 1 **REG ⊂ SREG, LIN ⊂ SLIN, CF ⊂ SCF.**

Theorem 2 **WKREG ⊆ SREG.**

Theorem 3 **WKLIN ⊆ SLIN.**

Theorem 4 **WKCF ⊆ SCF.**

Theorem 5 **SREG ⊂ SLIN ⊂ SCF.**

The hierarchy of families of languages is illustrated in Figure 1 where the solid arrows represent the proper inclusions of the lower families into the upper families, while the dotted arrow represents the inclusions.

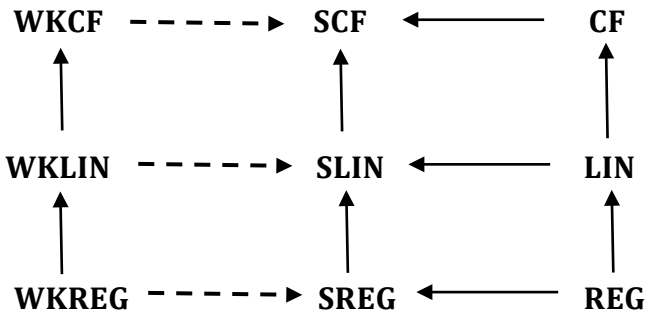

Figure 1. The hierarchy of families of static Watson-Crick, Watson-Crick, and Chomsky languages.

Based on Figure 1, the family of languages generated by regular, linear, and context-free grammars is denoted by **REG, LIN,** and **CF**, respectively. Besides, the notations **WKREG, WKLIN** and **WKCF** represent the family of languages generated by Watson-Crick grammars; while the notations **SREG, SLIN** and **SCF** represent the family of languages generated by static Watson-Crick grammars.

Conclusion

In this research, a new theoretical model known as the static Watson-Crick grammars is defined along with some of its computational properties. The static Watson-Crick grammars are classified into three types according to the Chomsky grammars, namely regular, linear, and context-free grammars. Referring to the results, the computational properties of the static Watson-Crick grammars is obtained through the relationship between the families of Chomsky languages and Watson-Crick languages. The findings show that the family of regular, linear, and context-free languages is strictly included in the family of static Watson-Crick regular, linear, and context-free languages, respectively. Besides, the family of Watson-Crick regular, linear, and context-free languages is included in the family of static Watson-Crick regular, linear, and context-free languages, respectively. In static Watson-Crick grammars, static Watson-Crick context-free languages have the highest computational power compared to both static Watson-Crick regular and linear languages. The results in this paper are useful for further research on the variant of static Watson-Crick grammars Watson-Crick, analysis of DNA structures, programming language and natural language structure.

Acknowledgement
This work was funded by the Ministry of Higher Education Malaysia (MoHE) under Universiti Teknologi Malaysia Shine Grant (UTMShine) Vote Number 09G89 and Universiti Teknologi Malaysia Fundamental Research Grant (UTMFR) Vote Number 20H70.

References

[1] Kari, L., Seki, S. and Sosík, P. 2012. DNA computing-foundations and implications. Handbook of natural computing. 3:1073-1128.

[1] Adleman, L. M. 1994. Molecular computation of solutions to combinatorial problems. Science. 266(5187): 1021–1024.

[2] Freund, R., Paun, G., Rozernberg, G., and Salomaa, A. 1997. Watson- Crick finite automata: DIMACS Series in Discrete Mathematics and Theoretical Computer Science. 297-327.

[3] Datta, S. and Mukhopadhyay, S. 2013. A composite method based on formal grammar and DNA structural features in detecting human polymerase II promoter region. PloS one. 8(2): e54843.

[4] Algwaiz A, Ammar R, Rajasekaran S. 2015. Framework for data mining of big data using probabilistic grammars. In *2015 Fifth International Conference on e-Learning (econf*). 241-246. IEEE.

[5] Subramanian, K.G., Hemalatha, S. and Venkat, I. 2012. On Watson-Crick Automata, In *Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology*. 151-156.

[6] Zulkufli, N.L.M., Turaev, S., Tamrin, M.I.M. and Azeddine, M. 2015. Closure Properties of Watson-Crick Grammars. In *AIP Conference Proceedings*. 1691(1): 040032.

[7] Paun, G., Rozenberg, G. and Salomaa, A. 1998. *DNA computing: new computing paradigms*. 1st edition. New York: Springer-Verlag Berlin Heidelberg.

[8] Linz P. 2006. *An introduction to formal languages and automata.* 5th Edition. Jones & Bartlett Learning.